Terminal Manager Services

For

Windows NT® & Windows 2000®

# Application Suite
# User's Manual

Control Module, Inc.
227 Brainard Road
Enfield, Connecticut 06082

## Disclaimer

*No part of this publication may be reproduced, or transmitted in any form or by any means without the written permission of Control Module, Inc. The software described by this publication is furnished under a license agreement, and may be used or copied only in accordance with the terms of that agreement.*

*This manual and associated software is sold with no warranties, either expressed or implied, regarding its merchantability or fitness for any particular purpose save for the warranty of the media. The information in this manual is subject to change without notice and does not represent a commitment on the part on Control Module, Inc.*

## License Agreement

*The software which accompanies this license (the "Software") is the property of Control Module, Inc., or its licensors and is protected by copyright law. While Control Module continues to own the Software, you will have certain rights to use the Software after your acceptance of this license. Your rights and obligations with respect to the use of this Software are as follows:*

*You may: (a) use one copy of the Software on a single computer; (b) make one copy of the Software for archival purposes, or copy the software onto the hard disk of your computer and retain the original for archival purposes; (c) use the Software on a SINGLE network, provided that you have a licensed copy of the Software for each computer that can access the Software over that network; (d) after written notice to Control Module, transfer the Software on a permanent basis to another person or entity, provided that you retain no copies of the Software and the transferee agrees to the terms of this agreement.*

*You may not: (a) sublicense, rent or lease any portion of the Software; (b) reverse engineer, decompile, disassemble, modify, translate, make any attempt to discover the source code of the Software, or create derivative works from the Software; or (c) use a previous version or copy of the Software after you have received a disk replacement set or an upgraded version as a replacement of the prior version.*

*This Agreement will be governed by the laws of the State of Connecticut. This Agreement may only be modified by a license addendum, which accompanies this license, or by a written document which has been signed by both you and Control Module. Should you have any questions concerning this Agreement, or if you desire to contact Control Module for any reason, please write to: Control Module, Inc., Customer Service, 227 Brainard Road, Enfield, CT 06082.*

# Trademarks

*Microsoft® is a registered trademark of the Microsoft Corporation.*
*Microsoft C® and Visual C++® are registered trademarks of Microsoft Corporation.*
*Microsoft Windows® is a registered trademark of Microsoft Corporation.*
*Microsoft Windows NT® is a registered trademark of Microsoft Corporation.*


©1996-1998 Control Module, Inc. – ALL RIGHTS RESERVED

# Table of Contents

**LICENSE AGREEMENT** ..... 1

**TRADEMARKS** ..... 1

**TABLE OF CONTENTS** ..... 2

**TABLE OF FIGURES** ..... 4

**REQUIREMENTS** ..... 5

HARDWARE CONFIGURATION ..... 5
SERVICE PACKS ..... 5
SECURITY ..... 5
DEPENDENCIES ..... 5

**TERMINAL MANAGER OVERVIEW** ..... 6

SYSTEM ARCHITECTURE ..... 6
MESSAGE SERVER ..... 7
JOURNAL SERVER ..... 7
NET SERVER ..... 7
NET MANAGER ..... 8

**QUICK START** ..... 10

INSTALLATION ..... 10
START-UP ..... 10
CONFIGURATION ..... 11

**OPERATION** ..... 12

NT SERVICES ..... 12
*Installation/Removal* ..... *12*
*Startup/Shutdown* ..... *12*
EVENT LOGGING ..... 13
MESSAGE SERVER ..... 13
JOURNAL SERVER ..... 14
*Live Transactions* ..... *15*
*Replayed Transactions* ..... *15*
*Database Management* ..... *15*
*Troubleshooting* ..... *16*
*Viewing the Journal Server Database* ..... *16*
CMINET SERVER ..... 17
*Connection Ports* ..... *17*
*Multidrop Networks* ..... *17*
*System Operating Modes* ..... *17*
*Terminal Configuration File* ..... *18*
*Clock Table* ..... 18
*Schedule Table* ..... 18
*Registry Settings* ..... *19*
CLIENT APPLICATIONS ..... 21
APPLICATION PROGRAMMER'S INTERFACE ..... 23
NET MANAGER ..... 24
*Interprocess Communications* ..... *24*


©1996-1998 Control Module, Inc. – ALL RIGHTS RESERVED

*Display Windows* ..... *26*
Network Configuration Tree View ..... 26
Show Config ..... 26
Show Comm ..... 27
Show MS/JS ..... 27
*Main Menu* ..... *28*
'Services' Submenu ..... 28
'Message Server' Submenu ..... 30
'Journal Server' Submenu ..... 32
'NetServer' Submenu ..... 34
'Device' Submenu ..... 38
'View' Submenu ..... 39
'Help' Submenu ..... 39
*Pop-Up Menus* ..... *39*
*Configuration List Menu* ..... *40*
*Port Pop-Up Menu* ..... *40*
*Terminal Pop-Up Menu* ..... *41*
*Adding a Terminal* ..... *42*
Clock Name ..... 42
Connection ..... 42
Terminal Address ..... 43
Auto Connection ..... 43
Schedule 'EDIT' Button ..... 44
Connection Schedule Dialog Box ..... 44
Daylight Savings Time ..... 44
Time Zones ..... 44
Download File ..... 44
Description ..... 45

**SUPPORT ..... 46**

UPDATES ..... 46
DIRECT SALES (SOFTWARE PRODUCT WAS PURCHASED DIRECTLY FROM CONTROL MODULE INC.) ..... 46
CMI VALUED RESELLERS (SOFTWARE PRODUCT WAS PURCHASED FROM QUALIFIED RESELLER) ..... 46


©1996-1998 Control Module, Inc. – ALL RIGHTS RESERVED

# Table of Figures


FIGURE 1 SYSTEM ARCHITECTURE ..... 9
FIGURE 2 DATA FLOW ..... 22
FIGURE 3 NETMANAGER INTERPROCESS COMMUNICATIONS ..... 25
FIGURE 4 NETMANAGER MAIN WINDOW: COMMUNICATIONS VIEW ..... 27
FIGURE 5 TEST NT SERVICE MANAGER LOCAL ..... 29
FIGURE 6 TEST NT SERVICE MANAGER REMOTE ..... 29
FIGURE 7 SERVICES START AND STOP CONTROL ..... 29
FIGURE 8 SERVICE CONTROL: ALL SERVICES RUNNING ..... 30
FIGURE 9 SERVICE AGENT CONNECTION DIALOG ..... 30
FIGURE 10 MESSAGE SERVER SUBMENU ..... 30
FIGURE 11 MESSAGE SERVER OPTIONS ..... 31
FIGURE 12 JOURNAL SERVER SUBMENU ..... 32
FIGURE 13 JOURNAL SERVER OPTIONS ..... 33
FIGURE 14 NETSERVER SUBMENU ..... 34
FIGURE 15 NETSERVER OPTIONS ..... 34
FIGURE 16 GLOBAL SET TIME SCHEDULE ..... 36
FIGURE 17 DEVICE SUBMENU ..... 38
FIGURE 18 SEND MESSAGE WINDOW ..... 38
FIGURE 19 HELP ABOUT ..... 39
FIGURE 20 CONFIGURATION LIST MENU ..... 40
FIGURE 21 TERMINAL POP-UP MENU ..... 41
FIGURE 22 ADD TERMINAL DIALOG ..... 42
FIGURE 23 CONNECTION SCHEDULE DIALOG ..... 44
FIGURE 24 SELECT DOWNLOAD FILE DIALOG ..... 45



©1996-1998 Control Module, Inc. – ALL RIGHTS RESERVED

# Requirements

## *Hardware Configuration*

CMI Terminal Manager Services for Windows NT® is designed to run on any of the following: NT 4.0 Workstation, NT 4.0 Server, Windows2000 Workstation, Windows2000 Server. Minimum processor, disk space and ram are detailed below:

| | **NT 4.0** | | **Windows 2000** | |
|---|---|---|---|---|
| | Minimum | Recommended | Minimum | Recommended |
| Processor: | Intel Pentium 166MHz | Intel Pentium 233MHz | Intel Pentium 166MHz | Intel Pentium 300MHz |
| RAM: | 32MB | 64MB | 128MB | 256MB |
| Disk space: | 100MB Available | 250MB Available | 100MB Available | 250MB Available |

NOTE:

The recommended configuration encompasses a standard Windows NT installation plus the Control Manager Terminal Manager product and one Terminal Manager Client application. It is the user's responsibility to determine system needs when other Client/Server or resource intensive applications are also resident on the machine.

## *Service Packs*

The Terminal Manager Services have been approved for use with Windows NT 4.0 Service Packs 3, 4, 5, and 6a. Service Pack 5 is the recommended minimum. Service Pack 6 is NOT approved for use with this product.

Terminal Manager Services version 2.0.0 and greater have been approved for use with Windows 2000 standard release, and Service Pack 1.

## *Security*

The installation of the Terminal Manager Services must be performed by a user who has local administrator privileges. Failure to assign adequate rights to the Terminal Manager user will have unexpected negative affects on the operation of the Terminal Manager Suite.

## *Dependencies*

The Terminal Manager Services rely on Remote Procedure Calls (RPC) for Inter-Process Communications. This functionality is provided through the "Remote Procedure Call Service" and the "Remote Procedure Call Locator Service". Both of these services must be installed and started automatically before the Terminal Manager Services will operate properly.

The CMI NetManager utility requires that the 'Server' networking service is installed and set for automatic operation. The NT 'Server' service is installed by default on NT Server systems, but must be manually installed on NT Workstation systems.


©1996-1998 Control Module, Inc. – ALL RIGHTS RESERVED

# Terminal Manager Overview

CMI's Terminal Manager Services for Windows NT® is a data collection and distribution package for use with Control Module's full line of data collection terminals. The package includes software for collecting data from Data Collection Terminals, storing the messages to permanent storage, and delivering the messages to one or more client applications. The client applications are able to login to the Terminal Manager Message Server and receive messages from the terminal network as well as other attached client applications. The client applications need not handle the device management tasks of the terminal network, or storage tasks of the raw transaction database. This package greatly simplifies the process of developing client applications to communicate with a network of CMI Data Collection Terminals.

Terminal Manager is provided with an Application Programmer's Interface (API) which is a set of dynamically linked libraries of functions used to work with the Terminal Manager Suite. The functions allow a programmer to Login to the Message Server, get and put messages to the Message Server, query the number of messages waiting in the Message Server and Logout of the Message Server as well as handle the replay of messages from the Journal Database Server.

The purpose of Terminal Manager is to provide users of CMI Data Collection Equipment with a package to collect and distribute data to customer designed and implemented applications. This package manages the details of:

- Collecting the data from the network
- Insuring data integrity
- Guaranteeing data storage for disaster recovery
- Filtering and delivery of messages to attached applications.

## *System Architecture*

The Terminal Manager Services Suite consists of three executable programs designed to run as Windows NT Services, plus supporting database files, dynamic link libraries (dlls), and a system management utility. The three services are: Message Server, Journal Server, and CMINetServer. Each one of these services performs specific functions necessary for the overall operation of the system. The service applications interface with each other via a library of functions. The underlying interprocess communications that support these functions is the native NT Remote Procedure Call (RPC) facility. In order for these services to properly communicate with each other, both the NT RPC Service, and the NT RPC Locator Service must be running.

There are two database files used by the system. The configuration of the user's data collection network is defined and maintained in a Microsoft Access© style database named 'CMINetServer.mdb'. The storage of raw data transactions collected from the data collection terminals is managed via a database engine embedded in the Journal Server service. The file name for the raw data transaction storage is 'journal.dbf'. This file requires the use of a companion index file 'journal.cdx'.


©1996-1998 Control Module, Inc. – ALL RIGHTS RESERVED

The Terminal Manager Services are configured and monitored by the CMI NetManager application. This application can be run either locally on the same machine as the Terminal Manager Services, or remotely, from any machine that has TCPIP access to the services machine.

## *Message Server*

The Message Server service is responsible for routing all data and command traffic between client applications and the data collection network. It maintains a list of all applications that 'login', and creates queues to hold the messages for each application until the application retrieves them. It is the single point of interface for all client applications that wish to collect the data terminal transactions, and send messages out to the data terminals.

The Message Server is the centerpiece of the Terminal Manager Suite. All of the clients and servers of the Terminal Manager Suite must login to the Message Server. In doing so, they are assigned a Terminal Manager application identification numbers (AppID) and two message queues are created for the application. All message traffic to and from the terminal network is routed through the Message Server. In this way a single message from the terminal network can be delivered to multiple client applications for processing

## *Journal Server*

The Journal Server service is responsible for storing, forwarding, and retrieving all records collected from the data collection terminal network. These records can be either data transactions, or terminal status messages. When the Journal Server receives a record to store in the database, it automatically forwards it to the Message Server who distributes the record to all connected client applications (Journal Server and NetServer are not considered client applications for these messages and will not receive them). The Journal Server also associates a serial number with the record as a unique identifier for that record when it is stored in the database.

The Journal Server service can also replay records from the database on demand. A request for a replay of records is made directly to the Journal Server from a client application via the Journal Server API (Application Programmer Interface) library of functions. The replayed records are forwarded to Message Server for delivery to the requesting client application only.

## *NetServer*

The NetServer service is the communication server for the data collection network. It reads the list of data collection devices and their connection information from the CMINetserver.mdb file. It can be configured to connect each data collection device either automatically or when a user initiates a manual connection request. It supports any combination of terminal network connections, via Ethernet, TokenRing, modem, or serial port. It can support one or more terminals through each connection. It provides the ability to configure the connections to run as a real-time data collection system, or as a batch system with scheduled data collection.

Once a connection is established using TCPIP, dialup, or direct serial communications, NetServer communicates directly to the data collection processor using the CMIPoll protocol for guaranteed data exchange.


©1996-1998 Control Module, Inc. – ALL RIGHTS RESERVED

The NetServer also incorporates the NetManager Agent which functions as a TCP connection server on port 3001, and facilitates the exchange of commands and status monitoring between the Terminal Manager Services and the CMI NetManager utility.

## *NetManager*

NetManager is the tool provided to manage the Terminal Manager Services and the data collection network. Unlike the services, this utility only needs to run when the user wants to set or view the configuration or status of any member of the Terminal Manager Services. It is designed to run locally on the same machine as the services or remotely, from any other machine with access to the services machine via TCPIP.

The services run independently of the NetManager and will continue to collect data from the network whether or not the NetManager is running.


©1996-1998 Control Module, Inc. – ALL RIGHTS RESERVED

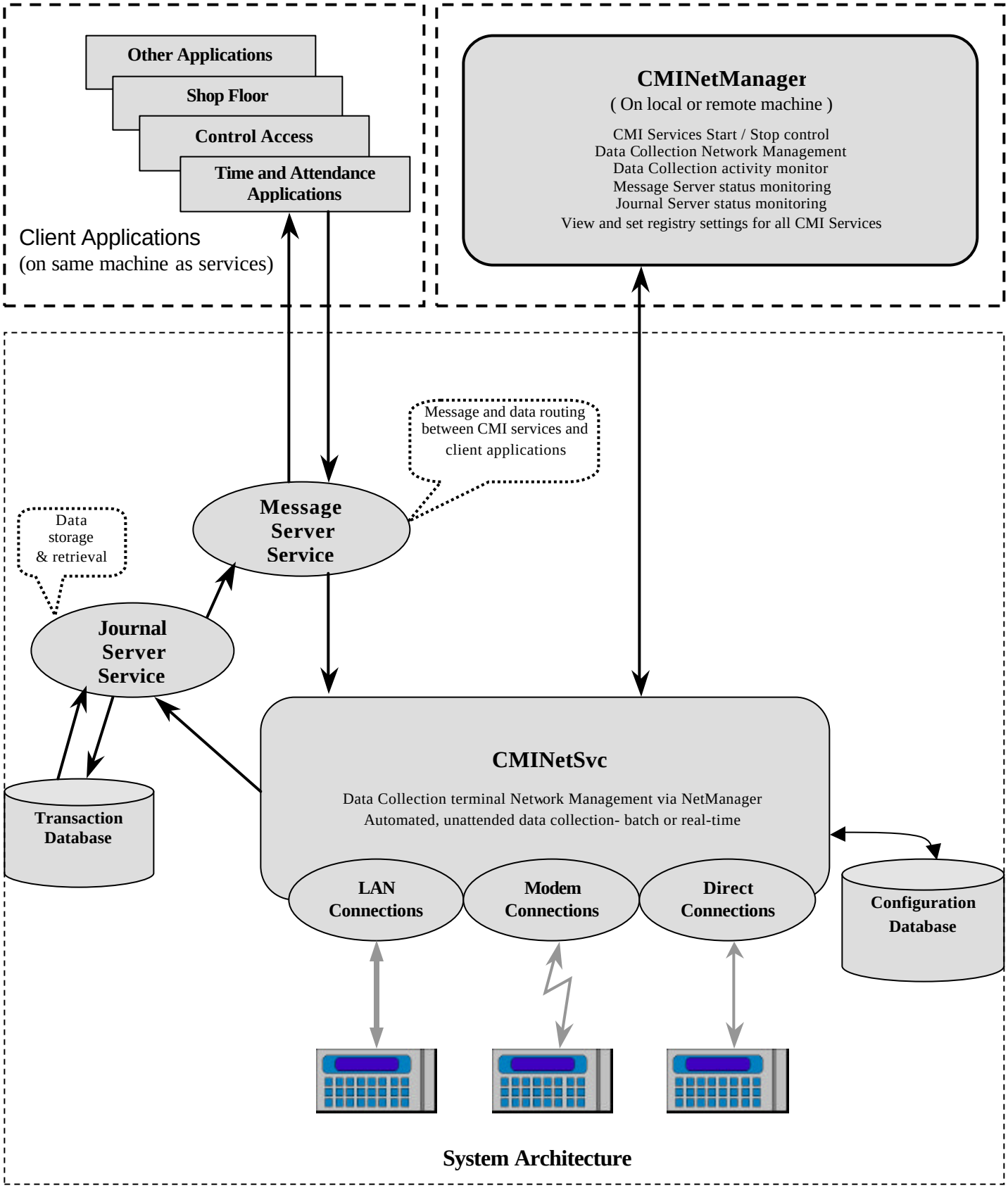



**Figure 1 System Architecture**



©1996-1998 Control Module, Inc. – ALL RIGHTS RESERVED

# Quick Start

This section of the User's Manual is for users who want to quickly install and operate the Terminal Manager product without the need for setup and maintenance of the Terminal Manager Suite. For additional information about individual components, their setup, maintenance and configuration please see the specific sections dealing with each individual server below. The following is a quick review of the installation process. More detailed information is contained in the 'CMI Terminal Manager Services Installation Manual'.

## *Installation*

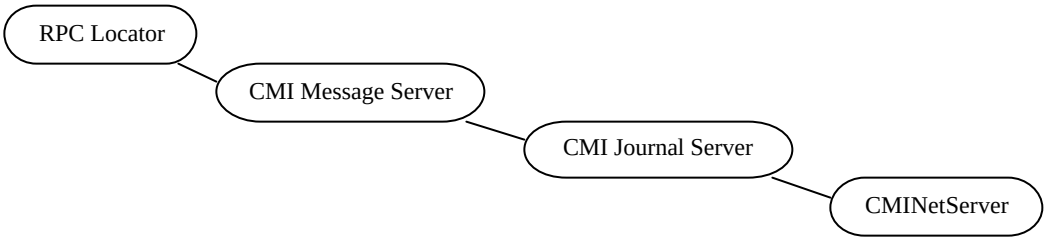
The Terminal Manager Services Suite is delivered on CDRom media. The setup utility will either automatically load when the CD is first recognized in the CDRom drive, or it can be run directly by the user by executing the 'setupex.exe' program located in the root directory of the CD.

The complete installation of the Terminal Manager Services actually takes place in three steps. The initial setup program will copy the service executables, support files, system files, plus a secondary installation utility for the CMI NetManager application to the target path. Before the initial setup utility completes, it launches a separate utility that registers the three CMI Service executables with the local NT Service Control Manager. Once the service registration completes, it passes control back to the initial setup utility. If the user intends to install the CMI NetManager utility on the same machine that is running the CMI services, the user can elect not to reboot, and then exit the setup utility. The user may then locate the secondary installation utility in the 'NetManager' folder of the installation target directory, and run 'setup.exe' to install the CMI NetManager before rebooting the system (selecting the 'Launch the program file' checkbox will automatically launch the NetManager installation). This secondary installation utility, if located in a shared folder, can be used to install the CMI NetManager from another machine accessible over the network for remote management of the CMI Terminal Manager Services. The NetManager installation utility is also available in the 'NetManager' folder on the installation CD.

Once all the desired installation utilities have completed, the system must be rebooted. Microsoft also recommends that the most recently installed service pack be re-applied.

## *Start-up*

CMI Terminal Manager Services are dependent on a specific start-up order. These dependencies are setup in the registry database by the CMI Services registration utility. The order of dependency is:





©1996-1998 Control Module, Inc. – ALL RIGHTS RESERVED

The CMI Services registration utility also sets each CMI Service to startup type ‘automatic’.

## *Configuration*

The CMINetServer service retrieves the data collection network configuration from the file ‘CMINetServer.mdb’. The initial installed copy of this file contains default device definitions. The user must customize the network configuration to include the actual installed data collection terminals. This customization can take place by either directly modifying the ‘CMINetServer.mdb’ file using the Microsoft ‘Access’ product, or by using the CMI NetManager utility. In either case, it is necessary to stop and restart the CMINetServer service to load the new network configuration.


©1996-1998 Control Module, Inc. – ALL RIGHTS RESERVED

# Operation

## *NT Services*

In order to maintain and troubleshoot the CMI Terminal Manager Services it is necessary to have a basic understanding of NT services. An NT service is fundamentally an application that is designed to perform specific tasks automatically without the need for direct user interaction. In the Windows NT operating system, this means that common functions that are user independent can be built into an application that runs when the system starts up, and before any user logs in. These service applications can be configured with the use of a separate application that can run as a standalone executable with a user interface, or as an applet that is recognized and executed via the NT Control Panel.

The CMI Terminal Manager Services have been designed to run as NT services so that the data collection functionality of communicating to data collection terminals, and the collecting and storing of data can be done without requiring a user login.

### Installation/Removal

Individual NT Services are usually installed by a setup program provided by the vendor of the service. These setup programs typically make calls directly to the NT Service Control Manager to register the application with NT Services. This is the same method used by the CMI 'InstallSvc.exe' program which executes after the CMI service executables have been copied to the system. Additionally, most service executables follow a convention that lets the user install and uninstall service applications individually without the necessity of writing a program that talks to the Service Control Manager. This method is embedded in most service applications, and is performed by executing the service application from a DOS window command line (or the Start/Run dialog), using the '–i' (install) or '–u' (uninstall) arguments. For example, to uninstall the CMI Message Server service, the user can open the NT Run dialog window, browse to locate the 'MessageServer.exe' program, and append '-u' before hitting the 'OK' button. It is more informative to perform the operation from the correct path in a DOS window so that all messages generated during the install or uninstall operation can be viewed in case an error occurs.

The NT Control Panel / Services utility does not allow for the installation of services, but only for the control of the startup and shutdown of the installed services.

### Startup/Shutdown

The user has control over starting and stopping NT services either through the NT Control Panel / Services utility, or through command line statements in a DOS window. The NT Services utility lets the user start or stop individual services, notifies the user of service dependencies when these operations are performed, and lets the user set the system startup mode of each service.

To start a service from the command line in a DOS window, the user must first know the name of the service as it is registered with the NT Service Control Manager. It is usually NOT the same name as the service executable itself. The name should be entered as listed in the Control Panel / Services utility dialog. The command used to start or start a service from a


©1996-1998 Control Module, Inc. – ALL RIGHTS RESERVED

command line is the NT 'net' command with the appropriate 'start' or 'stop' argument. For example, to stop the CMI Message Server service, the user would type- 'net stop "CMI Message Server" `. Note that it is not necessary to run the 'net start' or 'net stop' commands from the path of the service executable.

The CMI Terminal Manager Services, when installed, are configured to run automatically when the NT system starts up. If any of the CMI Services do not startup or shutdown properly, the log files generated by each of the applications should be checked for error messages.

For remote system monitoring and control, CMI provides the CMINetManager utility which provides start and stop control of CMI Terminal Manager services from a remote workstation.

## *Event Logging*

The three CMI Terminal Manager services all use the same convention to log events to a file. Each application maintains a daily log file of events. The name of each log file is created as the combination of a two-character identifier for the application that created the file, plus a six-character date code (mmddyy) for the day the file was created. 'MSmmddyy.txt' indicates a Message Server log file, 'JSmmddyy.txt' is a Journal Server file, and NetServer log files have the format 'NSmmddyy.txt'. If no events occur during the course of a 24-hour day, no event log file for that date will be created. Prior to version 1.1.0, the log files were created in the 'C:\winnt\system32' directory. The log files are now created in the '\log' subfolder of the path containing the CMI Terminal Manager Services executables. These files are normally small, typically less than 1Kbytes in size. However, if protocol debugging is turned on in NetServer, the NetServer log file can grow to be extremely large. Every day each application deletes the log file that was created 8 days ago to help restore disk space.

## *Message Server*

The bulk of the work performed by Message Server is the routing of messages between the Terminal Manager suite and client applications. The Message Server service consists of two pieces - the executable application 'MessageServer.exe', and the supporting dynamic link library 'MsgSrvClient.dll'. The application provides current status, and version information through the CMI NetManager utility.

Application developers building client applications in the 'C++' programming language will also need supporting application development files such as 'MsgSrvClient.lib', and the various 'include' files that are installed with the sample applications.

The most common failure for the Message Server is not having the required services installed and started before starting Message Server (see the Requirements section above). Once Message Server is running, it monitors its own incoming message queue for messages that need to be distributed to other client applications, and it also waits for login/logout requests from applications. The messages exchanged between applications through Message Server are passed in a structure that contains the message itself, plus other fields indicating the sequence number of the message, the origin of the message, and the destination of the message. ONLY messages that get to Message Server via Journal Server that contain actual data from the data collection network will have a sequence number associated with it. Commands exchanged between


©1996-1998 Control Module, Inc. – ALL RIGHTS RESERVED

applications are not logged by the Journal Server and do not have a sequence number. Message Server tracks the sequence number of data messages and records the sequence number of the last data message received by each client application. In this way, Message Server is able to help a client application 'recover' stored data transactions if the client application terminates and is restarted later while data collection continues unabated. Prior to Message Server Release F00227 Version 1.1.0, the last sequence number delivered to a client application was stored in the file 'c:\winnt\win.ini' under the client application name as 'LAST_SQNM = xxxx'. Versions 1.1.0 and later will store the sequence number of the last transaction delivered to each client application in the registry under the key:

'HKLM/Software/Control Module Inc/Message Server/(client name)'

When Message Server makes the transition from storing the sequence number in the .ini file to the registry, it will save the transferred number in the win.ini file under the client application name as 'SQNM_TO_REGISTRY = xxxx', and delete the 'LAST_SQNM=xxxx' entry.

There are two tunable Message Server parameters. They are the default inbound and outbound queue sizes that Message Server will use for each client application that logs in. These values are stored in the registry and can be viewed and modified by the user through the CMI NetManager.

## *Journal Server*

The Journal Server is a Client/Server style database server for permanently storing messages at the host before they are removed from the data collection terminal's message queue. It implements a self-purging, fast and efficient database for use by the CMINet Server and attached applications. The Journal Server stamps each new message with a sequence number ranging from 0 to 4,294,967,294. The Journal Server also includes a timestamp field for each record stored. If the record is a clock transaction, the timestamp field will be derived from the timestamp information in the transaction indicating the exact time the transaction was performed at the clock.

There are two files used by the Journal Server for transaction storage: the database file 'journal.dbf', and the index file 'journal.cdx'. If the data file 'journal.dbf' file does not exist, Journal Server will create a new empty file. However, if the data file exists, and the index file does not, Journal Server will not create it, and will be unable to properly store and forward the transactions. If both files do not exist, Journal Server will create both. It is extremely important to manage the two files as a pair. If the 'journal.dbf' is to be archived and restored at a later time for recovery of transactions, the associated index file must be saved and restored as well.

The Journal Server consists of two pieces- the executable application 'JournalServer.exe', and the supporting dynamic link library 'JrlSrvClient.dll'. Application developers building client applications in the 'C++' programming language will also need supporting application development files such as 'JrlSrvClient.lib', and various 'include' files that are installed with the sample applications.


©1996-1998 Control Module, Inc. – ALL RIGHTS RESERVED

## Live Transactions

CMINetServer uses a Journal Server API function to pass a data transaction to the database server via RPC and store it to disk. If the transaction storage operation completes properly, the successful function return allows the CMINetServer to complete the protocol exchange with the terminal by acknowledging receipt of the transaction. In this way, transactions are guaranteed to be either available in the terminal's transaction queue or stored to disk on the host computer. If the transaction storage function fails, data acceptance is not acknowledged by CMINetServer, and the terminal holds the transaction to resend it the next time the terminal is polled.

Once a live transaction has been stored to disk, it is forwarded to the Message Server for delivery to Terminal Manager client applications. Messages can be replayed from the Journal Server into the message queues of any of the attached client applications. These transactions are treated as ***live*** messages. All client applications that logged into Message Server with live transactions enabled will receive this transaction in their inbound queue.

## Replayed Transactions

The Journal Server will also replay messages from the transaction database. These replays can occur either as the result of a 'catch up' replay initiated when the client application logs in to the Message Server, or on demand, when the client application sends a replay request directly to Journal Server via the Journal Server API function calls. If the client application requests to receive a 'catch up' replay at login, the Message Server forwards the replay request to the Journal Server using the last known delivered transaction sequence number for that application as the starting point.

The client application can issue a replay request directly to the Journal Server at any time. When the Journal Server receives the replay request, it will extract the requested transactions and forward them to the Message Server for delivery to the requesting application. Replayed transactions are routed ONLY TO the requesting application. No other client applications will receive them. Transaction replay can be requested as a range of sequence numbers, or all transactions within a given start date/time and stop date/time.

In order to prevent queue overflow, the replay mechanism does not replay all requested transactions at once. It will replay up to 1000 transactions and then delay for 60 seconds before sending the next 1000, and will continue the cycle until all requested transactions have been sent.

## Database Management

The Journal Server automatically purges its database file once a day. At the schedule daily time the Journal Server purges old messages and reclaims the hard disk space used by these older messages. The time of day the purging activity takes place, and the maximum number of records that will remain in the database after the purge are registry values that the user can set through the CMI NetManager. The index file is also rebuilt when the purge takes place. The time of each purge activity is logged as an event in the Journal Server log file.

Like most other databases, record writes to the database cached and flushed at a regular interval. The default interval for the write flush is 5 seconds. This value can also be tuned by the user through NetManager.


©1996-1998 Control Module, Inc. – ALL RIGHTS RESERVED

## Troubleshooting

The Journal Server will report any errors in the startup operation to its log file. The most common failure for the Journal Server is not having the required services installed and started before starting Journal Server (see the Requirements section above). Because Journal Server must log in to the Message Server, the Message Server service must be started first.

Backup of the Journal Database used by the Journal Database Server is recommended. This backups need to happen while the Journal Server is NOT RUNNING. The files that need to be backed up are the database file "journal.dbf" and the index file "journal.cdx". These files are located in the 'C:\winnt\system32' directory.

## Viewing the Journal Server Database

The contents of the Journal Server database file 'journal.dbf' can be viewed using any third party tool that can access a 'FoxPro' style database. Viewing the database can be extremely beneficial in determining whether or not specific transactions have been received from the data collection network. It can also be used to get the transaction sequence number or date and time of specific transactions for recovery through a replay request from a client application.


©1996-1998 Control Module, Inc. – ALL RIGHTS RESERVED

## *CMINetServer*

The CMINetServer is the terminal communications hub for the Terminal Manager Services. All facets of communications with the terminal network are managed here. The CMINetServer is designed to simultaneously connect to CMI terminals through modems, through one or more of the host PC's serial ports or through a wide area network. Communications between CMINetServer and a terminal take place when a successful 'connection' is made to a terminal listed in the CMINetServer terminal configuration table. When CMINetServer is 'connected' to a terminal, it is then able to poll data from the terminal, and send messages to the terminal, until the connection is terminated. A 'connection' means that CMINetServer has both established a link with a 'port', and can successfully talk to the CMI terminal attached to that port.

### Connection Ports

CMINetServer supports three different connection port types: direct serial via a local COMM port, dialup via the Windows NT TAPI services, and LAN/WAN via TCPIP. The LAN/WAN connections can be either Ethernet or TokenRing. Direct serial connections are fixed to communicate at 9600 baud, 7 databits, and odd parity.

Once a link has been established with a port, the CMIPoll multi-drop protocol will be used to retrieve data from and send messages to one or all of the terminals configured to use to that port.

### Multidrop Networks

With the appropriate CMI hardware and network wiring, each connection type can support 1 to 32 CMI data collection terminals in a multidrop configuration. Contact CMI Technical support for assistance in the physical layout and setup of multidrop terminal network configurations.

Terminals installed in multidrop network configurations must have their communication parameters set the same as the connection type they are attached to. Serial multidrop networks communicate at 9600 baud, multidrop networks attached to a CMI LINC adapter for LAN/WAN connectivity must be set for 19.2 Kbaud, and modem multidrop networks must be set for the highest baud rate the CMI modem adapter allows (2400 baud or 9600 baud versions).

### System Operating Modes

CMINetServer provides three distinct modes of operation. These modes are: manual, automatic, and sequential. The manual mode is typically used when setting up or performing maintenance on the system. No terminals will be connected to automatically in this mode. The user manually connects to and disconnects from any terminal on demand. The automatic mode is intended to provide unattended data collection by automatically connecting to all terminals in the network configuration according to their individual connection setting. The sequential mode is also an unattended data collection mode that operates in a batch method, looping through the network configuration list, collecting data from one terminal at a time.


©1996-1998 Control Module, Inc. – ALL RIGHTS RESERVED

## Terminal Configuration File

The CMINetServer.mdb file is the file that contains all the configuration information CMINetServer needs to communicate to each data collection terminal in the data collection network. There are two tables in the database file: Clock (terminal descriptions), and Schedules (scheduled terminal connections and global functions). The phone numbers, baud rates, IP addresses, physical terminal addresses, download names, and alias names are all configured through the CMI NetManager and stored in this file. The terminal 'alias' field is used as a shorthand identifier for the terminal. Whatever name the user enters into the alias field will be passed to client applications along with the data as a unique identifier for the terminal that generated the data. There are additional fields for the default download script, time zone and daylight savings time information, and a description field for additional user information.

*Clock Table*

The 'Clock' table contains the configuration information for each terminal (aka 'Clock' in Time and Attendance applications) that CMINetServer will manage. The structure of the 'Clock' table is:

| Field Name | Type | Size | Description |
|---|---|---|---|
| ClockName | Text | 50 | 'alias' name for the terminal |
| Connection | Text | 20 | IP address, COMx, or telephone # |
| ContrlrAddr | Text | 1 | CMI Communication Controller address (default = 0) |
| TermAddr | Text | 2 | Terminal poll address (0 – 31) |
| DownLoadFile | Text | 80 | Path to download script file |
| Description | Text | 128 | Additional user information |
| Status | Text | 50 | Not used |
| ClockID | Text | 5 | Not used |
| Enabled | Yes/No | 1 | Yes = autoconnect, No = manual connect |
| AutoConnect | Yes/No | 1 | Yes = persistent connection |
| Scheduled | Yes/No | 1 | Yes = scheduled connection |
| DST | Yes/No | 1 | Yes = daylight savings time is observed |
| TimeZone | Text | 1 | 'E'astern, 'C'entral, 'M'ountain, 'P'acific, 'A'laska, 'H'awaii |

The 'TermAddr' field should contain the same terminal poll address that is displayed in the setup mode of the terminal. CMINetServer does not support CMI network controllers, so the 'ContrlrAddr' field should remain at the default value of '0'. The connection field should contain either an IP address in 'a.b.c.d' dotted notation, a serial communication port identifier valid for the machine (COM1, COM2, through COMxx if additional serial ports have been installed), or a telephone number (no alpha characters allowed). The remaining items will be discussed later in the NetManager section.

*Schedule Table*

The schedule table will contain an entry for each terminal that has been set for a 'scheduled' connection. There are also some system global functions that can be scheduled and therefore may be found in this table, like setting the time or downloading all terminals. There are 24 fields in the table, each one representing an hour of the day, from 'Hour00' to 'Hour23'.


©1996-1998 Control Module, Inc. – ALL RIGHTS RESERVED

## Registry Settings

Various operational parameters for CMINetServer are permanently stored as registry settings. All settings are stored as 'String Values'. The operational parameters that require customer configuration are all accessible through the NetManager NetServer/Options dialog. All other settings are usually modified under the supervision of CMI Technical Support if it is determined that a standard default needs to be modified for unexpected system conditions.
(Note: 'HKLM' = 'HKEY_LOCAL_MACHINE')

The standard CMINetServer options that are accessible via NetManager are:

HKLM\SOFTWARE\Control Module Inc\CMINetSvc\Startup\ConnectMode

Default value = 'MAN'

Acceptable values = 'MAN', 'AUTO', 'SEQ'

Description: The system 'Connection Mode' determines if the system will connect to the data collection devices in a manual or unattended mode of operation. For a detailed description, please refer to the section on CMINetServer Options in the CMINetManager section.

HKLM \SOFTWARE\Control Module Inc\CMINetSvc\Startup\Heart Beat Interval

Default value = '0' ( disabled, no message sent )

Acceptable values = '0' - '60' ( in minutes )

Description: CMINetServer will generate a 'HeartBeat' message that gets delivered to client applications by Message Server indicating that the system is alive and running. The message format is 'NS001', and it will be sent every 1 – 60 minutes depending on this setting.

HKLM \SOFTWARE\Control Module Inc\CMINetSvc\Startup\Auto Connect Interval

Default value = '60'

Acceptable values = '1' – '3600' ( in seconds )

Description: If the system 'ConnectMode' is set to automatic ('AUTO'), this value determines how frequently the system will attempt to restore connection to devices configured as 'Persistent' or 'Scheduled' connections.

HKLM \SOFTWARE\Control Module Inc\CMINetSvc\Startup\CMI Script Support

Default value = 'OFF'

Acceptable values = 'ON', 'OFF'

Description: Enables support for querying and storing the idle prompt of a device in order to place the device back online if left in the 'Locked' state.

HKLM \SOFTWARE\Control Module Inc\CMINetSvc\Startup\Clock Status Support

Default value = 'OFF'

Acceptable values = 'ON', 'OFF'

Description: Enables the generation of clock active ('NS3111'), degraded ('NS3116'), and disconnected ('NS3114'), messages to client applications.


©1996-1998 Control Module, Inc. – ALL RIGHTS RESERVED

The following operational settings are only accessible by directly editing the registry. They should only be changed by qualified NT support personnel after consulting with CMI Technical Support:

HKLM \SOFTWARE\Control Module Inc\CMINetSvc\Startup\Serial Delay
Default value = '500'
Acceptable values = '100' – '5000' ( in milliseconds )
Description: Sets the amount of time CMINetSvc will wait for a response to a protocol event from a device connected through a local serial port.

HKLM \SOFTWARE\Control Module Inc\CMINetSvc\Startup\Modem Delay
Default value = '1000'
Acceptable values = '100' – '5000' ( in milliseconds )
Description: Sets the amount of time CMINetSvc will wait for a response to a protocol event from a device connected through a dialup connection.

HKLM \SOFTWARE\Control Module Inc\CMINetSvc\Startup\Tcpip Delay
Default value = '3000'
Acceptable values = '100' – '5000' ( in milliseconds )
Description: Sets the amount of time CMINetSvc will wait for a response to a protocol event from a device connected through a TCPIP connection.

HKLM \SOFTWARE\Control Module Inc\CMINetSvc\Startup\TCP Connect Timeout
Default value = '15'
Acceptable values = '1' – '60' ( in seconds )
Description: Sets the amount of time CMINetSvc will wait to establish a TCPIP connection.

HKLM \SOFTWARE\Control Module Inc\CMINetSvc\Startup\Serial Contention Mode
Default value = 'OFF'
Acceptable values = 'ON', 'OFF'
Description: Enables or disables the Contention Mode portion of CMIPoll protocol to devices connection through a local serial port.

HKLM \SOFTWARE\Control Module Inc\CMINetSvc\NetManager\TCP Server Port
Default value = '3001'
Acceptable values = Any unused port number in the range '1024' – '65535'
Description: Allows the user to change the default connection port that NetManager will use to connect to CMINetSvc. This value must match the value entered in the NetManager/Services/Connect to Service Agent dialog.


©1996-1998 Control Module, Inc. – ALL RIGHTS RESERVED

## *Client Applications*

Terminal Manager is a server for the transactions collected from a CMI data collection terminal network. Client applications can register with the Message Server to receive the transactions and communicate with the data collection terminals. A client application may run either as a service or as a regular Windows NT application with a graphical user interface. Message Server can support up to eight client applications, not including the Journal Server and CMINetServer.

Client applications can also connect directly to the Journal Server in order to request the replay of transactions. When the Journal Server processes the replay command, it will extract the requested transactions from the database and forward them to the client application that made the request via the Message Server.

The following diagram illustrates the flow of data from the data collection terminals to the client applications.


©1996-1998 Control Module, Inc. – ALL RIGHTS RESERVED

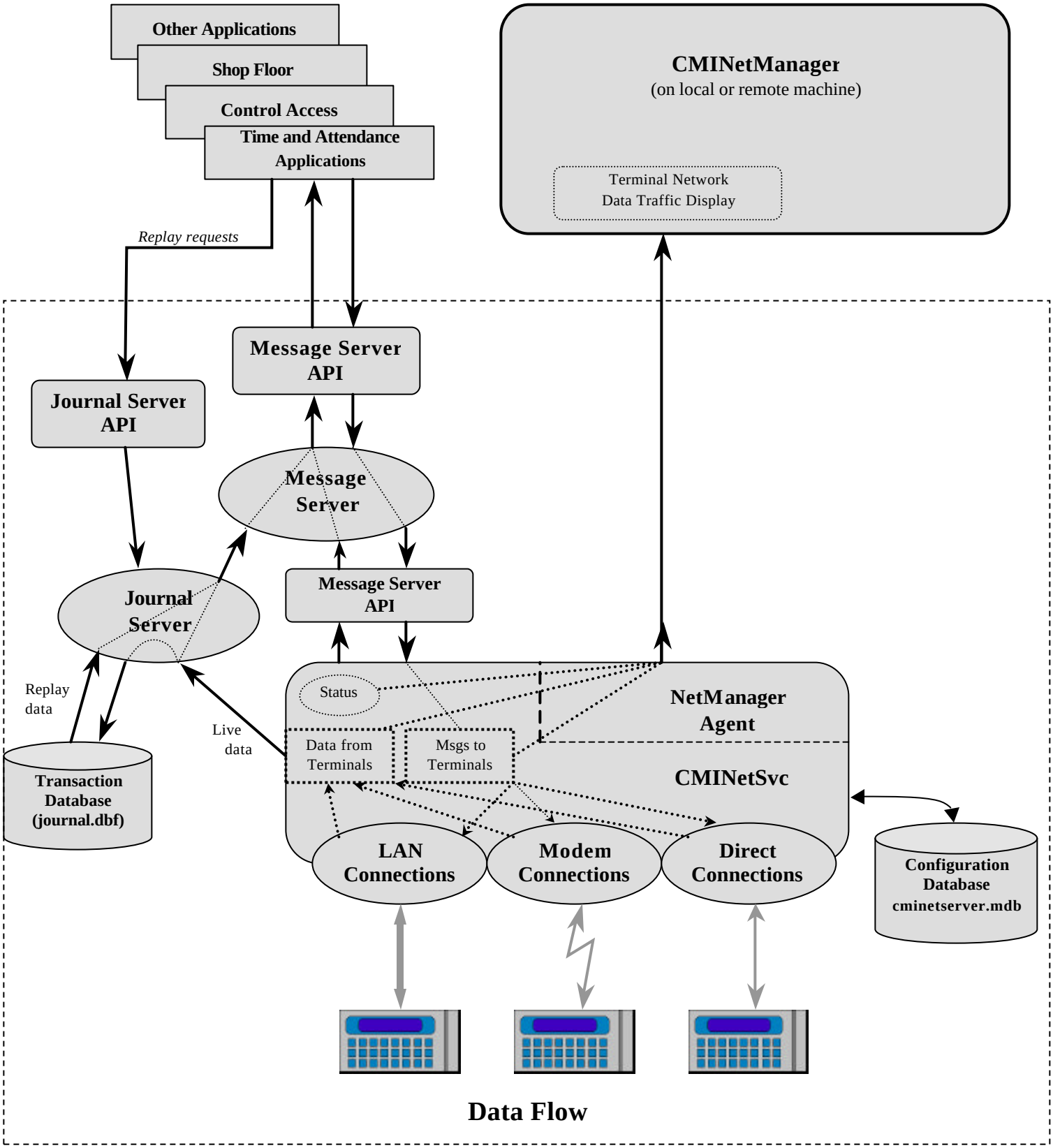



**Figure 2 Data Flow**


©1996-1998 Control Module, Inc. – ALL RIGHTS RESERVED

## *Application Programmer's Interface*

The Application Programmer's Interface is a set of dynamically linked libraries of functions used to work with the Terminal Manager Suite. The functions allow a programmer to Login to the Message Server, get and put messages to the Message Server, query the number of messages waiting in the Message Server and Logout of the Message Server as well as handle the replay of messages from the Journal Database Server. All client applications interface to the Terminal Manager Server with the use of the API libraries.


©1996-1998 Control Module, Inc. – ALL RIGHTS RESERVED

## *NetManager*

The CMI NetManager utility is designed to provide users with the ability to configure, control, and monitor the data collection activity performed by the CMI Terminal Manager Services. It is designed to run either on the same machine that the Terminal Manager services are installed on, or on any other workstation accessible to the Terminal Manager Server via TCPIP. The functions performed by the CMI NetManager fall into several categories-

- Service Control
  - Start and stop control of Terminal Manager Services
- Registry Settings
  - Viewing and setting registry items for the CMI services
- Network Configuration
  - Viewing and modifying the Data collection terminal network configuration
- Status & Activity Monitoring
  - Viewing the current status of Message Server
  - Viewing the current status of Journal Server
  - Viewing the current connection status of data terminal network
  - Monitoring data terminal network communications traffic

### Interprocess Communications

Several different interprocess communication methods are combined to provide the above functionality. Service start and stop control commands for all CMI services, plus requests for status from Message Server and Journal Server are implemented through the NT Service Control Manager. Status and monitoring activities are provided through a socket connection to the NetManager Agent embedded in the CMINetServer service. The NetManager Agent receives status information from the Message and Journal Services through a shared memory API. Reading and modifying the terminal network configuration is done through direct file access to the configuration database file. Reading and setting registry entries is accomplished via the socket connection to the NetManager Agent who performs the actual registry access functions.

The NetManager Agent is required to perform all functions except the starting and stopping of the CMI services. If the CMINetServer service is not running, NetManager will not be able to perform the full range of functions.


©1996-1998 Control Module, Inc. – ALL RIGHTS RESERVED

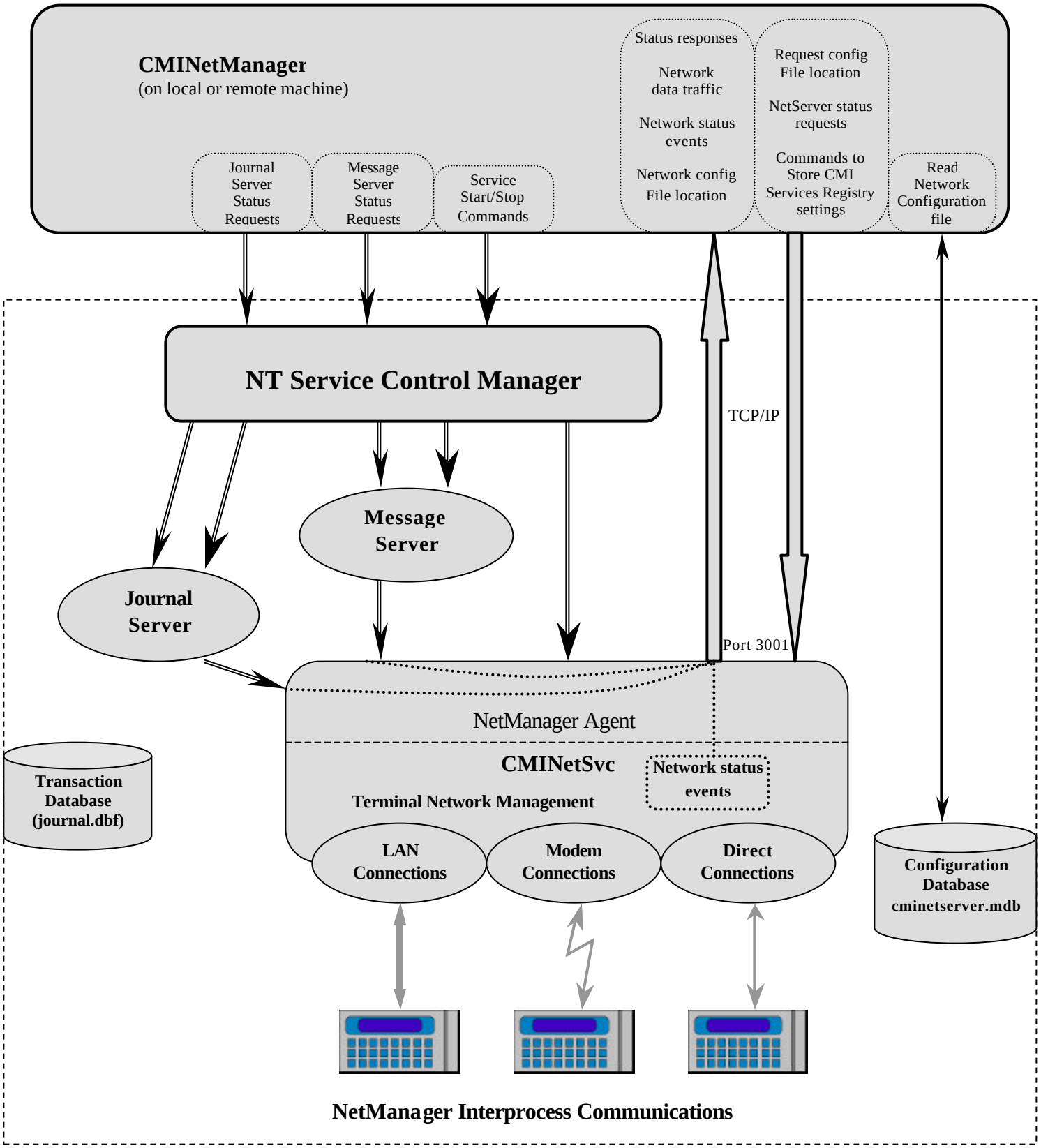



**Figure 3 NetManager Interprocess Communications**

©1996-1998 Control Module, Inc. – ALL RIGHTS RESERVED

## Display Windows

The NetManager display interface is made up of the standard title bar, a main menu, a row of status and command buttons, and two viewing window areas. There is one status button for each of the three CMI Services, and they are labeled 'MS', 'JS', and 'NS'. The color of these buttons indicates the running status of each one of the CMI Services respectively. Mauve means the service is stopped, green indicates the service is running, and yellow means a transition state between running and stopped. Each of these buttons can also be used to start/stop the service.

Each of the windows has an associated pop-up menu accessible by hitting the right mouse button when the mouse icon is in the window area. The pop-up menus vary from window to window depending on the functions that can be performed in the window. Most of the menus contain a 'Clear' function which is used to erase the current contents of the window.

The left viewing area is a fixed vertical window that always contains an explorer (tree) style display of the data collection network. There is a split bar on the right side of this window that can be used to increase or decrease the horizontal viewing area of the network tree. There is a preset minimum of space allocated for the network display, which prevents the split bar from moving too far to the left. The remaining window space is shared by swappable windows. Which set of windows is displayed in the shared area is selected by hitting one of the display command buttons above the display area. The display command buttons are: 'Show Config', 'Show Comm', and 'Show MS/JS'.

### Network Configuration Tree View

The network configuration tree view shows the configuration of the data collection network in an explorer 'file-tree' type display. The tree contains three levels of nodes. The first level is the primary or root node. The second level node represents a connection, or port through which the CMINetServer will communicate to a terminal. The third node is a terminal node. Each terminal node represents one record in the network configuration database. These nodes can be added, deleted, and edited via the pop-up menus.

The network tree view is also a dynamic display that shows the status of network connection activity graphically by changing the color of the port and tree nodes. A yellow node means a changing connection state- either connecting or disconnecting. A red node indicates a failure to connect, whereas a green node indicates the node is connected. Mauve is used to indicate a node configured to connect on a scheduled basis that is not currently connected. With the tree expanded and all nodes visible, the current network connection status is immediately visible.

### Show Config

When 'Show Config' is selected, the shared window area becomes one window that provides a list view of the individual data collection terminal configuration information. Which terminal configurations are shown in the window is determined by which level of the network configuration is highlighted in the left window network tree. The list does not get filled automatically when it is displayed. The user must highlight an item in the network tree, hit the right mouse button to bring up the pop-up menu, and then right click on 'Show'.

This window provides a quick view of terminal configuration. No items in the list can be edited or changed.


©1996-1998 Control Module, Inc. – ALL RIGHTS RESERVED

**Show Comm**

The 'Show Comm' displays two different sets of information concerning the data terminal network activity. There is a vertical window on the left side of the shared area that displays the connect and disconnect activity of the terminals as it occurs. This window will also show the CMINetServer version information as a result of selecting the NetServer/Version menu item.

The remaining window space shows the actual data and message activity taking place between Terminal Manager and all of the data collection terminals. This space can be viewed as one window showing the sends and receives together in sequential order as they occur. It can also be split horizontally into two windows, with the messages being sent out to the terminal network in the upper window, and the data coming in from the network in the lower window. The user can toggle between the two different types of network traffic displays by positioning the mouse within the area, right clicking to bring up the pop-up menu, and then right clicking on 'Merge' (if two windows are showing), or 'Split' (if one window is showing).

**Show MS/JS**

The third shared window display option is the 'Show MS/JS' window. When selected, the shared area will be split into two horizontal windows used to display the results of status queries issued to Message Server and Journal Server via the main menu.

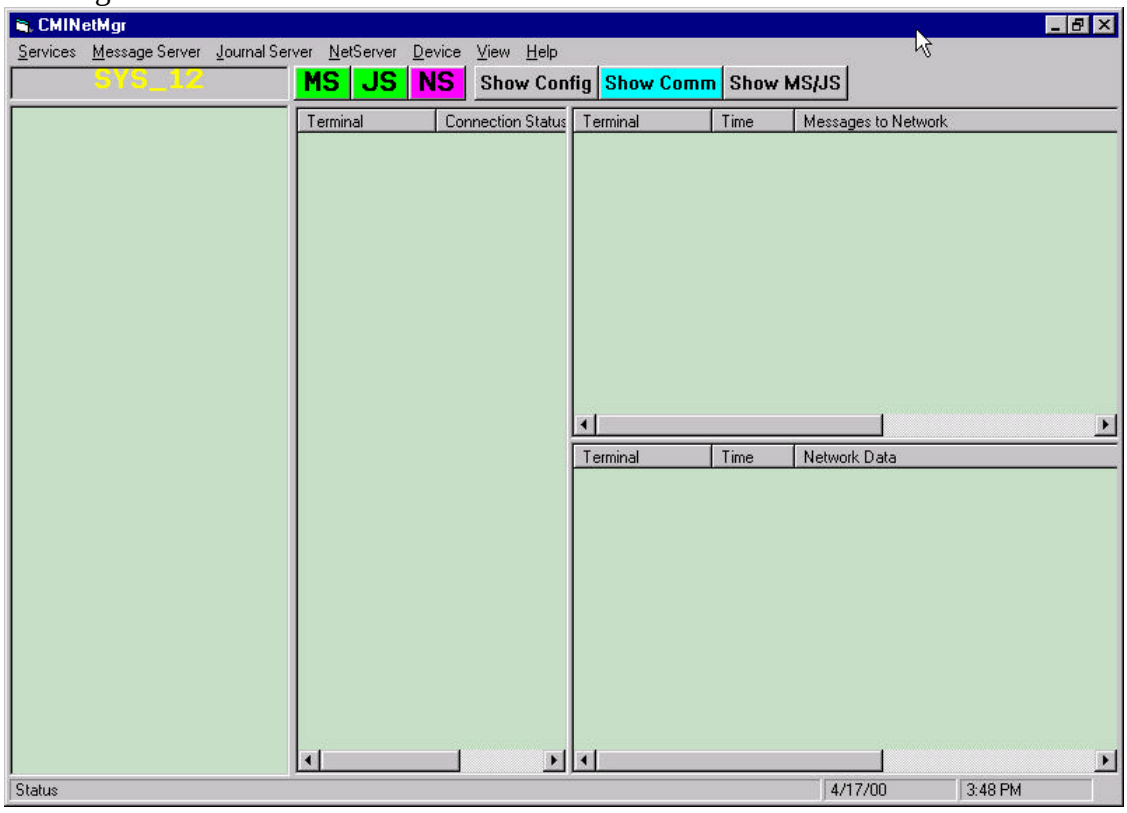



**Figure 4 NetManager Main Window: Communications View**

Figure 4 display details:

- Computer Name of Terminal Manager Services = SYS_02
- Message Server service running
- Journal Server service running
- CMINetServer service not running


©1996-1998 Control Module, Inc. – ALL RIGHTS RESERVED

- 'Show Comm' window display selected
- Network communications displayed in split windows
- Computer name in yellow indicates no connection to NetManager Agent
  (because CMINetServer is not running)
- No network tree in left window
  (because no connection to NetManager Agent)

## Main Menu

The main menu is always displayed below the title bar of the NetManager application. The functions that can be executed through the main menu are organized in groups from left to right. The first group of functions is the service control functions. They are accessed as a submenu through the 'Services' main menu item. Then comes the 'Message Server, 'Journal Server', 'NetServer', and 'Device' submenus. The organization of the main menu items from left to right is loosely based on the order with which they are utilized to manage the system operation. The service control functions are necessary to connect with the NT Service Control Manager before the service start/stop control is available. Once the three services are running, it is possible to view Message Server, Journal Server, and CMINetServer status items on demand through their respective menus. It is also possible to read and modify the current options for each of the services through their respective 'Options' menu item. When all three services are running, and the terminal network is displayed in the network tree window, it is possible to perform the terminal functions found in the 'Devices' submenu.

The main menu item 'Exit' does not have a submenu. Clicking on this menu item will terminate the NetManager application. It is available in the menu to provide a keystroke method of closing the application. The application may also be terminated with a mouse click on the application close icon ('X') in the title bar.

### 'Services' Submenu

In the 'Services' submenu, the 'CMI Services Location' item is used to identify the computer where the CMI Services are running. There are only two choices. The services are either running on the same machine as NetManager, or on a different machine. If the services are not running on the same machine as NetManager, the user must identify the machine where the CMI Services are located by entering either the IP address or computer name of the machine. Once the user provides the information, NetManager will connect to the NT Service Controller on the designated machine and try to locate the CMI Services and display their running status.


©1996-1998 Control Module, Inc. – ALL RIGHTS RESERVED

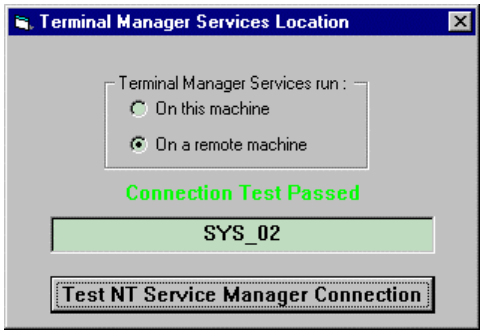
Terminal Manager Services Location
Terminal Manager Services run :
On this machine
On a remote machine
Connection Test Passed
Test NT Service Manager Connection

**Figure 5 Test NT Service Manager Local**

The figure at left shows that the Terminal Manager Services are running on the same machine as NetManager. Hitting the test button will cause NetManager to attempt a connection with the NT Service Control Manager. If the test succeeds, the 'Connection Test Passed' message will be displayed in green. If unable to access the NT Service Control Manager, a 'connection failed' message will be displayed in red.

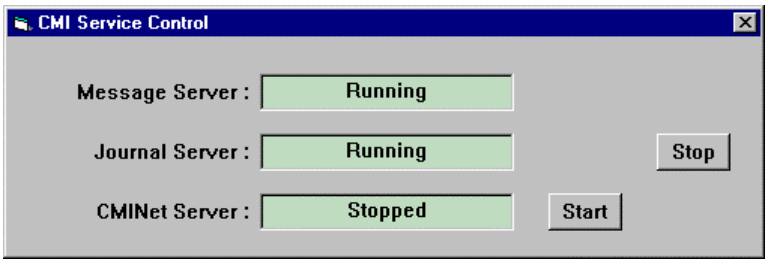
If the CMI Services are running on a remote machine, the user must enter either the IP or computer name of the remote machine in the edit box. Then the connection test should be performed to make sure the NT Service Control Manager is accessible on the remote machine.

**Figure 6 Test NT Service Manager Remote**




Once a connection has been established to the NT Service Control Manager, NetManager will check for the three CMI Services and change the color of the MS, JS, and NS buttons according to their status. In the previous illustration of the application, the MS and JS icons were green, and the NS icon was mauve. In this example, the next step is to start the NetServer service. Selecting the 'Service Control' item of the Services submenu will display the following dialog box.

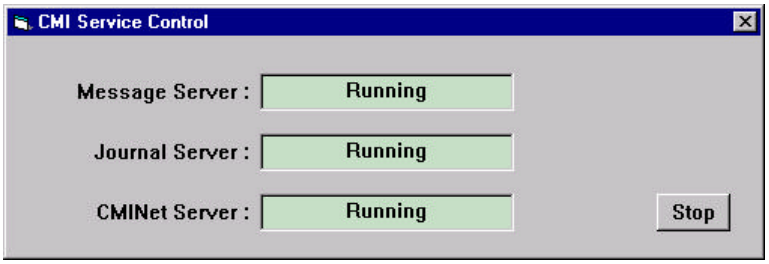
**Figure 7 Services Start and Stop Control**




This dialog box provides the start and stop control for the CMI Services. It is designed to limit the start and stop activity to the proper sequential order: Message Server, Journal Server, and CMINetServer. In this example, Message and Journal Servers are already running. The user is permitted to either complete the startup of the services by hitting the start button to start NetServer, or to shut down the services, if desired, by stopping the Journal Server. If the user hits the 'Start' button the CMINetServer service will start, and the dialog box will update both the status of the service, and which command buttons are available to the user.


©1996-1998 Control Module, Inc. – ALL RIGHTS RESERVED

**Figure 8 Service Control: All Services Running**

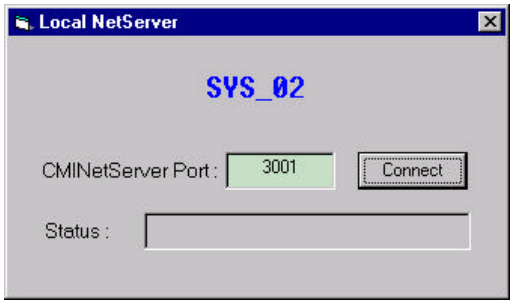



Normally, all three of the services should be started automatically by NT when the system boots up. The ability to start and stop CMI services from NetManager is provided as a backup tool for support purposes.

The final step to establish full interaction between NetManager and the CMI Services is to test the TCPIP connection between NetManager and the NetManager Agent resident in the NetServer service. This is done by selecting the final Services submenu item 'Connect to Service Agent'. Selecting this menu item will bring up the following dialog box.

The title bar of the dialog indicates that the CMI Services are running on the local machine (the same machine as NetManager). The computer name of the machine is 'SYS_02'.

**Figure 9 Service Agent Connection Dialog**




Hitting the connect button will attempt to establish a TCPIP connection with the NetManager Agent in CMINetServer through port 3001. If the connection test passes, the dialog box will immediately disappear, and the computer name will be displayed in green above the network tree window. If the connection test fails, an error message will appear in the Status box.

Port 3001 is the installation default port number used by NetServer and should not be changed unless there is a conflict with another application. If a conflict does occur, the port number can be changed by performing the following steps-

- Stop the CMINetServer service
- Modify the registry entry 'HKEY_LOCAL_MACHINE/SOFTWARE/Control Module Inc/CMINetSvc/NetManager/TCP Server Port' to the desired port number
- Start the CMINetServer service
- Change the port number in the NetManager Services/Connect to Service Agent dialog
- Test the connection

## 'Message Server' Submenu

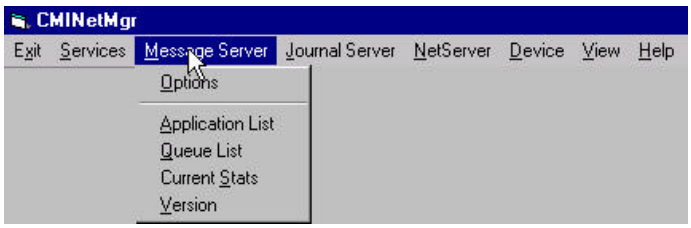


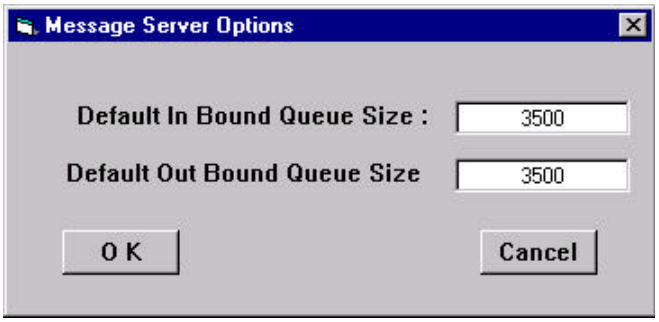

**Figure 10 Message Server Submenu**


©1996-1998 Control Module, Inc. – ALL RIGHTS RESERVED

Selecting the Message Server menu 'Options' item will bring up a dialog box containing the user customizable options used by the Message Server service. Initially, the values for each option item will be empty. NetManager will send a command via the NT Service Control Manager to Message Server requesting the settings currently in use by the application. Message Server will pass the response to the NetManager Agent who then forwards the information to NetManager. If NetManager does not receive a response to the request after 5 seconds, it will display an error message box, and the options dialog will be closed.

Once the information has been retrieved, the edit boxes will be filled in with the current values. The user modifiable options in Message Server are the default inbound and outbound

**Figure 11 Message Server Options**




queue sizes. There is also a checkbox to enable debugging messages which turns on the logging of informational events to the Message Server logfile for troubleshooting purposes. If any items in the dialog are changed, the user can either 'Cancel' to close the dialog without accepting the changes, or hit 'OK' to accept and store the changes. The changes are forwarded from NetManager to the NetServer application. The NetServer application will store the values in the registry. The new values will not take effect until the Message Server service is restarted.

The remaining menu items all generate a request to the Message Server for specific status information. The request is passed to Message Server through the NT Service Control Manager. The responses are sent to the NetManager Agent in the NetServer service and forwarded to NetManager through the TCP connection. All the responses will be displayed in the Message Server window of the shared window area.

The 'Application List' command will generate a response that identifies all applications connected to the Message Server, including the Journal Server and NetServer services. The list appears as follows:

**** Applications List ****

| AppNum | AppName | Logged In |
|---|---|---|
| 00 | Journal Server | Yes |
| 01 | CMINet Server | Yes |
| 02 | None | No |
| 03 | None | No |
| 04 | None | No |
| 05 | None | No |
| 06 | None | No |
| 07 | None | No |
| 08 | None | No |


©1996-1998 Control Module, Inc. – ALL RIGHTS RESERVED

The 'Queue List' command response will show the list of connected applications and the current status of their inbound and outbound message queues.

**** Queue List ****

| Num | Name | IB Size | IB Count | OB Size | OB Count |
|---|---|---|---|---|---|
| 00 | Journal Server | 0100 | 000 | 5000 | 000 |
| 01 | CMINet Server | 3500 | 000 | 3500 | 000 |
| 02 | None | 000 | 000 | 000 | 000 |
| 03 | None | 000 | 000 | 000 | 000 |
| 04 | None | 000 | 000 | 000 | 000 |
| 05 | None | 000 | 000 | 000 | 000 |
| 06 | None | 000 | 000 | 000 | 000 |
| 07 | None | 000 | 000 | 000 | 000 |
| 08 | None | 000 | 000 | 000 | 000 |

The InBound count is the number of messages waiting to be read by the application in the 'Name' column. The OutBound count is the number of messages that have been placed in the queue by the application that need to be picked up by Message Server for deliver to other clients. If the InBound count is greater than 0 when there are few or no data collection transactions being generated at the terminals, it may be an indication that the client application is not servicing the inbound queue fast enough or at all.

The 'Current Stats' request will generate a display of the total message activity through Message Server.

Total messages received and delivered: xxxxx
Total bytes received: xxx kbytes
Total bytes transmitted: xxx kbytes

The 'Version' command will return the release and version numbers used by CMI to track both the Message Server application, and the associated client dynamically linked library (MsgSrvClient.dll).

CMI Message Server: Release F00227 Version x.x.x
Dynamically Linked Library: Release F00227 Version x.x.x

**'Journal Server' Submenu**

The Journal Server menu items function in the same manner as the Message Server menu items. There is an options dialog that lets the user modify customizable settings. There are status requests that generate a response that will be seen in the Journal Server window of the shared window area. There are also two commands to the Journal Server that will perform routine database functions and display the results of the activity.

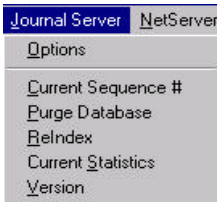



**Figure 12 Journal Server Submenu**


©1996-1998 Control Module, Inc. – ALL RIGHTS RESERVED

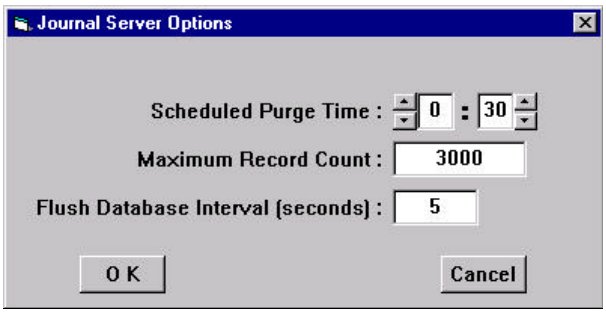



**Figure 13 Journal Server Options**

The Journal Server options dialog shows three user modifiable settings. The first is the time of day that the automatic daily purge of the database takes place. The user should set this time for a time of day when little data collection activity will be occurring since writes to the database will be held off until the purging has completed. The second field is the maximum numbers of records that will remain in the database after the purge. This value varies from system to system. CMI recommends that this number be set to the number of transactions collected during one 24 hour period times the number of days (plus one) that the user wants to have available for possible replay requests. When the purge takes place, the oldest records above the maximum record count will be deleted from the database. The final option, 'Flush Database Interval' is the number of seconds between each database record cache write. The flushing interval is intended to optimize the performance of the database write activity. These values, if changed and accepted, will be sent to NetServer. NetServer will store the new values in the registry. The Journal Server service will have to be restarted for the new values to become effective.

Journal Server tracks each record stored in the Journal database with a unique sequence number. The user can view the current sequence number at any time by selecting the 'Current Sequence #' menu item and viewing the results in the Journal Server window.

The Journal database can be purged on demand through the 'Purge Database' menu item. Selecting this item will send a command to the Journal Server service through the NT Service Control Manager which will initiate the purge. The start of the purge cycle is indicated in the Journal Server window with a timestamped message:

'Mm/dd/yyyy hh:mm Purging Database'

When the purge completes, the results of the activity are displayed. The results indicated whether or not the database had reached the maximum number of records, if so, how many records were purged, and the current record count in the database. Note that there is no relationship between the current sequence number and the number of records in the database. If the maximum record count has not been reached, the message will be:

Database has not reached threshold.
Records in database: xxxxx

If the maximum record count has been exceeded, the message will be:

Removed xxxx records from Database Journal

The re-index database function can also be performed on demand through the Journal Server submenu. It is usually not necessary to re-index the database manually unless there has been an interruption in the NT system operation. The re-index function is also performed automatically when the database is purged.


©1996-1998 Control Module, Inc. – ALL RIGHTS RESERVED

Selecting the 'Current Statistics' menu item will result in the following display.

Records in database: xxxxx
Max database records: xxxxxx
Scheduled purge time: hh:mm AM (PM)
Disk write interval: xx

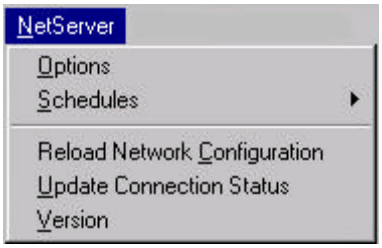
The 'Version' command will return the release and version numbers used by CMI to track both the Journal Server application, and the associated client dll (JrlSrvClient.dll).

| | | |
|---|---|---|
| CMI Journal Server: | Release F00228 | Version x.x.x |
| Dynamically Linked Library: | Release F00228 | Version x.x.x |

## 'NetServer' Submenu

The NetServer submenu, like the Message and Journal menus, has an options dialog for setting application parameters. Unlike the other menus, there are no status functions that display results in the shared window area. Only the 'Version' item will generate a response that will appear in the shared window area. The remaining functions all operate in conjunction with the live display of network status in the fixed network tree window.

**Figure 14 NetServer Submenu**




The 'Connect Clocks' group in the Options dialog sets the System Connection Mode of operation for the data terminal network.

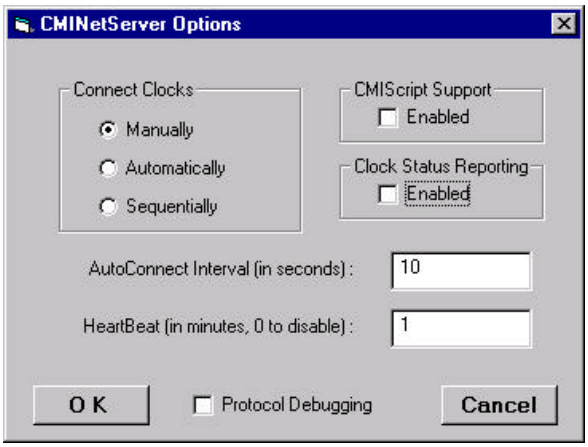



**Figure 15 NetServer Options**

The System Connection Mode is the single most significant option that is set by the user. It determines how CMINetServer will handle the connection to ALL the devices in the configuration list as a whole. There are three types of connection modes: manual, automatic, and sequential.

The manual mode is typically used for testing, demonstration, development, and maintenance. When in this mode, no automatic connections will be made to any device in the configuration list regardless of the setting in the 'Auto' column of the list. The user has total control of connecting and disconnecting terminals individually.


©1996-1998 Control Module, Inc. – ALL RIGHTS RESERVED

The automatic mode is designed to provide unattended, continuously maintained connections to ALL devices in the configuration list which have the 'Auto' field set to 'Y'. In addition, any device that has an 'Auto' setting of 'S' for scheduled, will be connected to when the system determines that the user defined connection schedule indicates that a connection should be made. Scheduled devices are most commonly used for, but not restricted to, devices that connect via a modem. When a scheduled device is connected in automatic mode, the connection is only maintained as long as data is being retrieved from the device. When the device has no more data to send, the connection is terminated. If the 'Auto' field of a device is set to 'Y', but the connection to that device has been lost (possibly loss of network communications), the system will attempt to automatically reconnect to that device when the 'Auto Connect Interval' has elapsed. If the user desires a 'batch' mode data collection system, all devices can be setup with an 'Auto' field of 'S' and set up with individual connection times.

The Sequential mode of operation is an optional method of implementing 'batch' style data collection. In this mode, the system will step through the configuration list one device at a time, checking the 'Auto' field. If a device is configured as 'Y', a connection will be made long enough to collect all the transactions from the device. When all the transactions have been collected, the connection is terminated, and the system steps to the next device in the list. The system will scroll through the list from top to bottom continuously, and then return to top of the list to continue the cycle. If a device is setup as a 'S'cheduled connection, the system will check to see if there is a current or recent schedule event that needs to be serviced. If so, the connection is made, data is collected, the connection is terminated, and the system steps to the next device in the configuration list. If there is not a current or recent schedule event to be serviced, the device is skipped, and the system advances to the next device in the list. The sequential mode of operation is useful for batch oriented data collection, utilizes less system resources than the automatic mode in terms of open connections, and will provide continuous collection from all devices on a regular, rotating basis.

The 'Autoconnect Interval' setting is only used in automatic mode. It sets the value of a timer that runs a check on device connections at the specified interval. If a terminal is configured to be connected automatically, and the connection has been lost, NetServer will attempt to restore this connection every time the interval occurs.

The 'Heartbeat' is also a timer setting. It is used to send a message at a regular interval to the Message Server indicating that the CMINetServer is running. Client applications, by watching for this message, can determine that the Terminal Manager system is operational. The text of this message received by the client applications is 'NS001'. Setting the value to 0 disables the generation of the heartbeat message.

The 'protocol debugging' box, if checked, enables the generation of informational messages concerning the protocol exchange between NetServer and the terminal network. These messages will be logged to the NetServer log file. Enabling the protocol debugging should be done only when necessary to determine the cause of a communications problem. The frequent communications activity that takes place will result in an extremely large log file when this option is enabled.

The CMI Script Support and Clock Status Reporting options are features supported by CMINetServer version 2.0.0 and later. If CMIScript Support is enabled, CMINetServer will query each device, when connected, with the '6F01' and 'S03' status request messages. The '6F01' request will return the device's idle prompt number ( loaded by the script file ). CMINetServer will store the idle prompt information for sending to the device in event that the


©1996-1998 Control Module, Inc. – ALL RIGHTS RESERVED

device is left in the 'Locked' mode. The 'S03' response indicates the current operating state of the clock- locked, interactive, or at a prompt.

Clock Status Reporting, when enabled, will cause CMINetServer to send the following messages to Message Server for delivery to all client applications:

- 'NS3111' – device has connected
- 'NS3114' – device has been disconnected on demand
- 'NS3116' – device connection has been terminated due to failure of device to respond

The three menu items 'Get Configuration File', 'Refresh List', and 'Update Status' all affect the display of the network tree in the fixed window. Their operation will be discussed on the section describing the contents of the network tree display. The remaining menu item 'Version' sends a command to NetServer through the TCP connections and receives the following response which will be displayed in the 'Connection Status' window of the 'Show Comm' display.

NetServer Release: F00229
NetServer Version: x.x.x

The Schedules submenu contains menu items for setting the schedule of global system functions. Currently, only the Set Time function is supported as a global function. Selecting the 'Set Time' menu item will bring up a schedule dialog box similar to the one used to schedule connections for terminals.

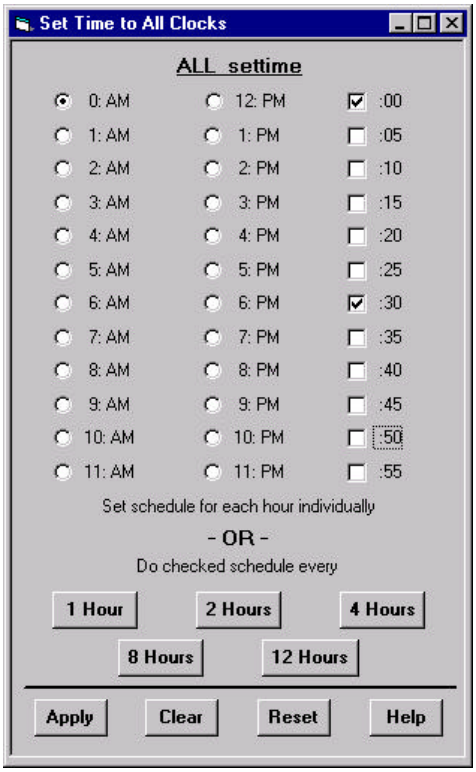



**Figure 16 Global Set Time Schedule**

The set time schedule dialog box is used to set the time of day when CMINetServer will send the current system to all connected devices. The schedule is a daily schedule, and will be


©1996-1998 Control Module, Inc. – ALL RIGHTS RESERVED

used every day of the week. The schedule has a resolution of 5 minutes. The user may set up ANY combination of times throughout the entire 24 hour period. The schedule dialog box has one radio button for each hour in a day. Only one of these radio buttons can be selected at a time. When an hour is selected, the checkbox list of the 5 minute intervals displays which times have been selected for that hour. The user may check any number of these boxes to select the times when the set time command will occur. The user can select each hour individually and set up when the command will occur during that hour, or the user can set the command schedule for the displayed hour and then repeat those command times throughout the entire 24 hour period by hitting one of the repeat buttons for every: 1, 2, 4, 8, or 12 hours. The entire schedule can be erased with the 'Clear' button. The user can accept the new schedule by hitting the 'Apply' button, or revert to the last saved schedule with the 'Reset' button.


©1996-1998 Control Module, Inc. – ALL RIGHTS RESERVED

**'Device' Submenu**

The 'Device' submenu provides a list of functions that can send commands to one or more data terminals. How many terminals receive the command is determined by the currently selected item in the network tree display. If the 'Configuration List' item is selected, the selected command will be sent to all connected terminals in the network. If a connection node is currently selected, the command will be sent to all terminals that connect through that port. If a terminal node is selected, the command will be sent to the highlighted terminal only. Selecting one of the menu items either with a mouse or keyboard will send a message to NetServer indicating the command to be sent and the scope of the command delivery. NetServer is responsible for building the command in the format that will be recognized by the terminal and delivering it to each terminal identified by the scope of delivery.

**Figure 17 Device Submenu**

The 'Set Time' menu item will send the current system time to the selected terminal or terminals. The terminals that receive the command will immediately set their internal clock to the new date and time value, and if the date or time is currently being displayed, the display will be updated also.

The 'Status Check' command sends a message to the network that must be responded to by each terminal receiving the message. The responses will appear in the communications window as an incoming message.

The 'Download' command will initiate the downloading of the CMI Command Script file associated with the destination terminal in the network configuration. Download files can range from a few to hundreds of lines in size. For this reason, the scope of delivery of the download command is limited to a terminal or a connection. Selecting this menu item with the 'Configuration List' node highlighted in the network tree will not initiate any downloads.

The 'Send Message' item will bring up a dialog box that lets the user enter the message to be sent to the terminal. The dialog box contains the message entry area, a 'Send' button to launch the message, a 'Clear' button to erase the contents of the message window, and two character buttons used to embed ASCII control characters used by the CMI Command Script into the message text. When the 'Send' button is hit, the contents of the message text box will be sent to NetServer. NetServer will then forward the message to one or more terminals as indicated by the scope of delivery in the 'To:' window.

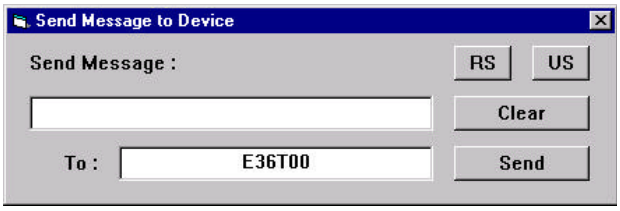



**Figure 18 Send Message Window**


©1996-1998 Control Module, Inc. – ALL RIGHTS RESERVED

**'View' Submenu**

The 'View' submenu contains only one item which lets the user show or hide the status bar at the bottom of the application viewing area. Each time the 'Status Bar' item is hit, it will toggle the viewing state of the status bar. If the item is displayed without a check mark, then the status bar is currently hidden, and clicking on the 'Status Bar' menu item will show the window status bar and place a check mark to the left of the menu item.

**'Help' Submenu**

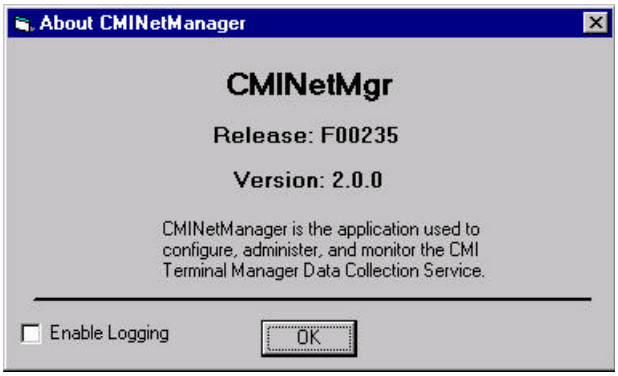



**Figure 19 Help About**

The 'Help' submenu contains only one item. The menu item 'About CMINetMgr' follows the Microsoft Windows standard convention of displaying application information to the user. The 'About' window for NetManager displays the application's executable name, plus the CMI release number and version of the executable. If logging is enabled, all messages exchanged between CMINetManager and CMINetServer will be put in the NetManager log file.

## Pop-Up Menus

Various windows in NetManager can access a pop-up menu via the right mouse button. The menu that appears will contain items that will perform a function for the entire window, or a highlighted member in a window list control.

All the windows in the shared screen area have a pop-up menu containing the item 'Clear'. Selecting the 'Clear' function will clear the contents of the window, and new entries will be displayed starting at the top of the window.


©1996-1998 Control Module, Inc. – ALL RIGHTS RESERVED

## Configuration List Menu

The fixed window on the left side of the application area is the Configuration List (or network tree) window. This window displays the data collection network configuration as a tree view similar to the Windows Explorer display. Hitting the right mouse button while the cursor is within this window will bring up a menu of functions that can be performed on the currently highlighted item.

There are three levels of nodes that are displayed in this window. The root node is simply called the 'Configuration List' and represents a common point for all devices contained in the data collection network. Highlighting this node, and clicking the right mouse button will bring up a menu that contains a mix of functions that work as commands to all devices in the configuration list, and functions that change the way the network configuration is displayed.

Show Config
Displays the 'Show Config' window in the shared window area and displays a record for each device in the network configuration.

Expand
Fully expands all nodes in the network tree so that all devices can be viewed.

Collapse
Collapses the tree to display just the connection nodes.

Add Device
Brings up an 'Add Clock' dialog for entering a new data collection device description to the network.

**Figure 20 Configuration List Menu**

The five menu items listed in the bottom group of functions are executed for ALL terminals in the configuration list. They are standard Terminal Manager functions for connecting terminals, disconnecting terminals, checking terminal status, setting the terminal time to the current system time, and downloading the CMI Script command file.

## Port Pop-Up Menu

Right clicking in the Configuration List window with a port node highlighted will display a pop-up menu of functions. There are two groups of functions in the menu. One group consists of network configuration functions. The bottom group is the same as the group of terminal functions found in the Configuration List menu.

The network configuration functions are: Show, Add, and Delete. Selecting the 'Show' function will cause the display of the 'Show Config' network configuration window populated with only records associated with the currently highlighted port in the network tree. The 'Add' function will bring up the 'Add Clock' dialog with most parameters defaulted to the same values as the first terminal description attached to the port. The user will only need to enter an alias, a description, and set the value of the Terminal Address to complete the entry. Note that the 'Add' function always adds a full terminal description to the configuration. It does not add just a port, since the port is part of the terminal description.


©1996-1998 Control Module, Inc. – ALL RIGHTS RESERVED

The Delete function will delete all the terminals associated with the selected port. When the configuration tree is re-displayed, the port node will no longer exist since there will be no terminals associated with the deleted port.

The five menu items listed in the bottom group of functions are executed for all terminals associated with the selected port in the configuration list. They are standard Terminal Manager functions for connecting terminals, disconnecting terminals, checking terminal status, setting the terminal time to the current system time, and downloading the CMI Script command file.

## Terminal Pop-Up Menu

The items in the terminal pop-up menu are the same as the items in the port menu, with one exception. The ability to 'Add' a terminal description to a specific port is only available through the port menu. The 'Add' function has been replaced with the 'Edit' function in the terminal menu. Selecting the 'Edit' function from the terminal menu will bring up the clock edit dialog box with all the fields filled in from the configuration of the highlighted terminal. It is not possible to edit the 'Clock Name' field through this function. In order to change the name of a terminal, it is necessary to delete the existing definition and 'Add' a new terminal from the port menu.

**Figure 21 Terminal Pop-Up Menu**

Selecting the 'Show' function will display the configuration of the single highlighted terminal in the configuration list window. Selecting 'Delete' will delete the current terminal from the configuration, as well as the port, if there is only on terminal defined for the port.

The bottom five menu items listed will be executed only on the highlighted terminal.


©1996-1998 Control Module, Inc. – ALL RIGHTS RESERVED

## Adding a Terminal

Selecting the 'Add' item from the pop-up menus will bring up a dialog box with all the fields necessary to describe and configure the new terminal in the network configuration. The user must enter all fields in the 'Add' clock dialog or else an error message will appear. The user may use the 'Cancel' button at any time to abort the changes and exit the dialog. The Terminal Alias is the name the rest of the Terminal Manager Suite will use when addressing a message to this terminal. In this way the attached applications do not need to know if this terminal is connected via LAN, modem or direct connect. The administrator must also complete the Comm Port field, the CMIPoll Address, the Download File, and the Description fields. The CMIPoll Address is the physical terminal address entered in the test mode of the CMI data collection terminal. The CMIPoll Address and the terminal address must be the same in order to comminute with the terminal. Physical addresses MUST be unique among all of the CMI terminals attached to a Comm Port, modem, or TCPIP connected device.

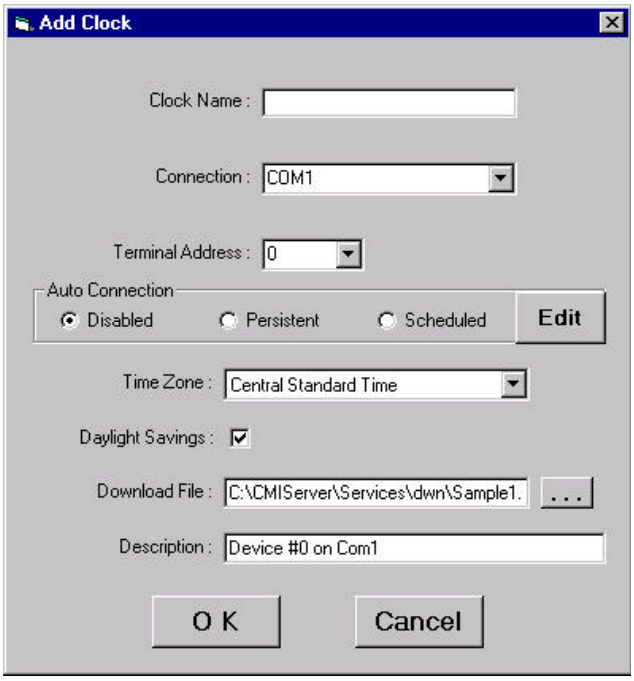



**Figure 22 Add Terminal Dialog**

**Clock Name**

The user should enter a short string in this field to describe the location and/or function of the terminal. This field must be unique for every terminal in the configuration database.

**Connection**

CMINetServer supports Control Module data collection devices connected directly to the serial port of the host Windows NT machine, through a dialup connection, and connected to the LAN via TCP/IP (Ethernet and Token Ring). CMINetServer automatically distinguishes the connection type based on the entry in the connection type field. Any entry using the 'a.b.c.d' dotted notation of IP addressing is assumed to be a network connection. Leading zeros in the four fields should be suppressed, otherwise, if present, may make the connection attempt fail. A direct serial connection is identified by either typing in 'COMx' (x = the serial port number), or selecting a 'COM' entry from the integrated pull down list. All other entries in this field are assumed to be dialup types (e.g. 18005551212).

CMINetServer supports the two basic serial port connections COM1 and COM2. Shared interrupts are not supported. Additional serial ports can be added to the system with the use of third party asynchronous communication adapters that provide Windows NT driver support as ports COM3 – COMxx. Up to 32 CMI terminals may be attached to a single serial port in a multidrop RS-485 configuration.


©1996-1998 Control Module, Inc. – ALL RIGHTS RESERVED

Users are responsible for the proper setup and configuration of the networking infrastructure. Proper communications to any new CMI network connected terminals should first be tested using the 'ping' utility. Up to 32 CMI terminals may be connected through a single TCP/IP connection in a multidrop RS-485 configuration. Up to 256 simultaneous TCP/IP connections can be maintained.

Dialup connections to CMI terminals equipped with CMI modems is achieved through the native Windows NT Modem interface. Some properties of the modem connection used by CMINetServer must be set to specific values through the Windows NT Control Panel/Modems setup utility. The connection preferences must be set to: 7 data bits, odd parity, and one stop bit. Call connection timeout should be set to 90 seconds. In the Advanced Connection Settings, error control must be disabled (unchecked). Hardware flow control may or may not be disabled. Software flow control should not be enabled.

**Terminal Address**

This field is the poll address of the terminal device that is connected through the communication port. Set this address to the address shown in the terminal setup display.

**Auto Connection**

When the CMINetServer is set to a system mode of maintaining connections to terminals automatically, or on a rotating (sequential) basis, it looks at the auto connection mode of each terminal to determine if a connection should be made. The 'Auto Connection' box in the configuration dialog has three modes displayed, but only one can be enabled at a time.

The 'Disabled' mode indicates that CMINetServer will not try to automatically establish a connection to the terminal. This setting is useful for locations that have been allocated, but not yet installed, or taken out of service for operational reasons.

The 'Persistent' mode means that CMINetServer, when in either of the two automatic system collection modes, will automatically connect to the terminal. If the system mode is set for 'Automatic', and the terminal connection is lost, CMINetServer will attempt to re-connect on a regular interval based on an internal system timer. The 'Persistent' mode is indicated by a 'Y' in the 'Auto' column of the configuration list. Terminals connected to a modem should never be setup in persistent mode.

The 'Scheduled' mode is used to connect to a terminal only at specified times. The scheduled mode is most often used for, but not limited to, modem communication types. When a terminal set for scheduled connection is automatically connected to, the connection terminates when all data has been retrieved from the terminal. A terminal designated as 'Scheduled' will automatically connect when the schedule dictates, when the system is in either 'Automatic' or 'Sequential' mode. A scheduled connection that is made in either of these two modes is only connected until all outbound messages are sent to the terminal, and all transactions have been collected from the terminal. When the system is in 'Manual' mode, the user can connect and disconnect to 'Scheduled' terminals at will.


©1996-1998 Control Module, Inc. – ALL RIGHTS RESERVED

## Schedule 'EDIT' Button

The 'EDIT' button to the right of the 'Auto Connection' box is only enabled when a terminal is set to the 'Scheduled' connection mode. When this button is hit, a dialog box for setting the schedule of connection for the terminal being edited is displayed.

## Connection Schedule Dialog Box

The connection schedule dialog box is used to set the time of day when CMINetServer in an automatic system collection mode will establish connection to the terminal. The schedule is a daily schedule, and will be used every day of the week. The schedule has a resolution of 5 minutes. The user may set up ANY combination of collection times throughout the entire 24-hour period.

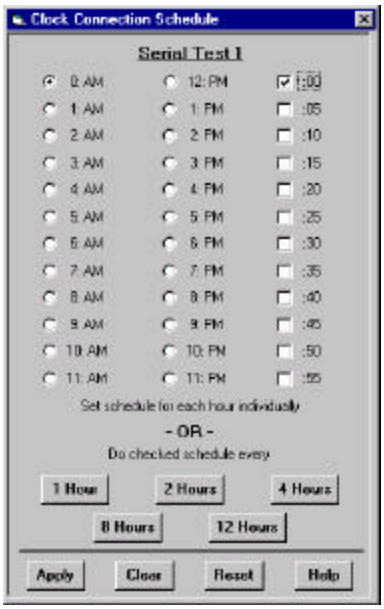



**Figure 23 Connection Schedule Dialog**

The schedule dialog box has one radio button for each hour in a day. Only one of these radio buttons can be selected at a time. When an hour is selected, the checkbox list of the 5 minute intervals displays which times have been selected for that hour. The user may check any number of these boxes to select the times when automatic connection will occur. The user can select each hour individually and set up which connection times will occur during that hour. Optionally, the user can set the connection times for an hour and then repeat those connection times throughout the entire 24 hour period by hitting one of the repeat buttons for every: 1, 2, 4, 8, or 12 hours. The entire schedule can be erased with the 'Clear' button. The user can accept the new schedule by hitting the 'Apply' button, revert to the last saved schedule with the 'Reset' button.

## Daylight Savings Time

This checkbox should be checked if the terminal is in a location that observes daylight savings time. It is not an indication whether or not daylight savings time is *currently* in effect.

## Time Zones

The terminal configuration lets the user identify the time zone that the terminal is located in, and whether or not daylight savings time is observed at the terminal location. All set time commands issued to the terminal will be adjusted based on these settings relative to the current system date and time of the CMINetServer machine. The user may select one of the time zones from the pull down list. The time zones supported are all time zones defined in the NT registry of the machine running the Terminal Manager Services.

## Download File

The user can associate the terminal with a file containing CMI Script for download to the terminal. There is a button that will bring up a file selection dialog box


©1996-1998 Control Module, Inc. – ALL RIGHTS RESERVED

that looks for files with the default extension '.dwn'. Each terminal may have a different file, or all terminals could use the same file. The file selection is dependent upon the user application. If the path to the selected file is unavailable when the user attempts to initiate a download, an error message will be displayed indicating that the file could not be found.

The download file itself should contain each CMI Script command on a separate line encapsulated by double quotes ( e.g. the line "1E" (including quotes) will be parsed as a command for the terminal, and the character string '1E' will be extracted and sent to the terminal. Lines without double quotes are considered comment lines.

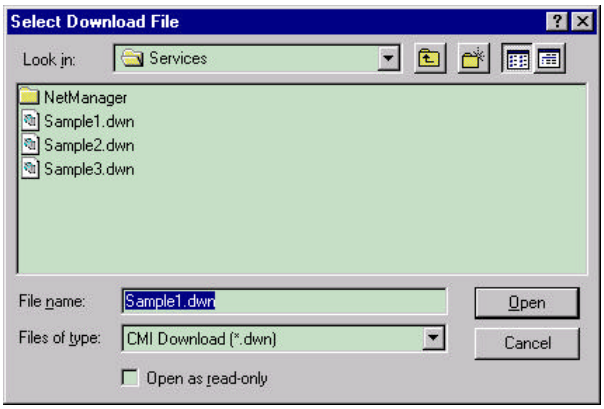



**Figure 24 Select Download File Dialog**

## Description

The description field is for user information only. The user may use this field to provide additional description of the location or function of the terminal.

***IMPORTANT NOTE:***

***When adding or editing terminals, it is necessary to run the system in manual mode and disconnect all connected terminals. After the adding and editing is complete, the NetServer service must be stopped and restarted.***


©1996-1998 Control Module, Inc. – ALL RIGHTS RESERVED

# Support

## *Updates*

CMI provides free updates of the software for one year from the date of the original purchase. The updates are not automatically distributed. The customer must contact CMI with the serial number and version of their original product to receive the update. Version and update information concerning the product can be obtained by contacting 'systems@controlmod.com' and providing the name, version, build code, and serial number from the installation CD label.

## *Direct Sales* *(Software product was purchased directly from Control Module Inc.)*

Control Module provides 90 day free technical support for the installation and operation of the software starting on the date of the first software support incident. The free technical support expires one year after date of delivery if the 90-day support period has not been initiated.
To initiate the 90-day free support period, contact the CMI Systems Engineering Department with the serial number from the installation CD at one of the following numbers.

Systems Engineering
Control Module, Inc.
800-722-6654
860-745-2433

For support options beyond the initial 90-day support period, contact the Control Module Sales Department at (800) 722-6654 or (860) 253-6222.

## *CMI Valued Resellers* *(Software product was purchased from qualified reseller)*

CMI Valued Resellers are knowledgeable and equipped to provide the initial support of the software installation and use. Customers who purchased the product from a CMI Valued Reseller should contact their representative for support. Extended support can be contracted through the Valued Reseller, or, upon agreement, directly from Control Module Inc.


©1996-1998 Control Module, Inc. – ALL RIGHTS RESERVED